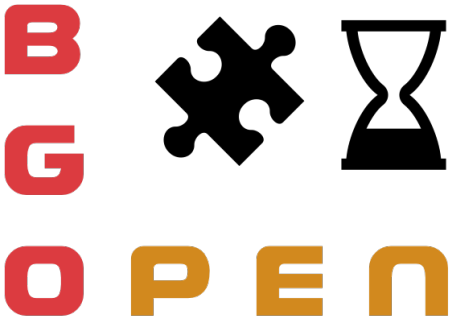
Bergen Open 2019

November 2, 2019



Problems

A Alehouse
B Bus Ticket
C Climbing Stairs
D Drive Safely
E Equilibrium
F Fence Bowling
G Great GDP
H Howl
I Ice Cream
J Jane Eyre
K Killing Chaos

Do not open before the contest has started.

Advice, hints, and general information

- The problems are not sorted by difficulty.
- Your solution programs must read input from *standard input* (e.g. `System.in` in Java or `cin` in C++) and write output to *standard output* (e.g. `System.out` in Java or `cout` in C++). For further details and examples, please refer to the documentation in the help pages for your favorite language on Kattis.
- For information about which compiler flags and versions are used, please refer to the documentation in the help pages for your favorite language on Kattis.
- Your submissions will be run multiple times, on several different inputs. If your submission is incorrect, the error message you get will be the error exhibited on the first input on which you failed. E.g., if your instance is prone to crash but also incorrect, your submission may be judged as either "wrong answer" or "run time error", depending on which is discovered first. The inputs for a problem will always be tested in the same order.
- If you think some problem is ambiguous or underspecified, you may ask the judges for a clarification request through the Kattis system. The most likely response is "No comment, read problem statement", indicating that the answer can be deduced by carefully reading the problem statement or by checking the sample test cases given in the problem, or that the answer to the question is simply irrelevant to solving the problem.
- In general we are lenient with small formatting errors in the output, in particular whitespace errors within reason. But not printing any spaces at all (e.g. missing the space in the string "`1 2`" so that it becomes "`12`") is typically not accepted.

  The safest way to get accepted is to follow the output format exactly.
- For problems with floating point output, we only require that your output is correct up to some error tolerance.

  For example, if the problem requires the output to be within either absolute or relative error of 10^{-4}, this means that

  - If the correct answer is 0.05, any answer between 0.0499 and $.0501$ will be accepted.
  - If the correct answer is 500, any answer between 499.95 and 500.05 will be accepted.

  Any reasonable format for floating point numbers is acceptable. For instance, "17.000000", "0.17e2", and "17" are all acceptable ways of formatting the number 17. For the definition of reasonable, please use your common sense.

Problem A
Alehouse
Problem ID: ../alehouse

Public Domain, Tavern Scene by David Teniers the Younger, via Wikimedia Commons

Making friends can seem impossible, but going to the alehouse makes it easy — it is actually the only way to make friendships. Luckily, the alehouse is extremely good at its task: if two people are inside simultaneously, they instantly become friends. People even become friends if they meet each other in the door as one leaves and one enters the alehouse!

In Consistentville, each of its n residents goes to the alehouse exactly once each week, and always during the same milliseconds as the week before. This is convenient for everyone, since then nobody needs to befriend new people all the time, which can be quite exhausting.

You are contemplating a move to Consistentville in order to adopt their well-ordered lifestyle, and have decided that you want as many friends as possible. However, you don't actually enjoy ale that much, so you decide to limit your weekly visit at the alehouse to at most k milliseconds. What is the maximum number of friends you can get?

Input

The first line of input contains two positive integers n ($1 \leq n \leq 100\,000$), and k ($0 \leq k < 604\,800\,000$). The next n lines describe at which millisecond each of the original residents of Consistentville enters and leaves the alehouse every week. Specifically, the i^{th} line consists of two integers a_i and b_i ($0 \leq a_i \leq b_i < 604\,800\,000$) indicating that the i^{th} resident enters the alehouse at millisecond a_i and leaves the alehouse at millisecond b_i each week.

Output

A single integer, the maximum number of friends you can get.

Sample Input 1	Sample Output 1
6 2 0 2 1 8 5 9 2 4 7 8 10 10	4

This page is intentionally left blank.

Problem B
Bus Ticket
Problem ID: ../busticket

Dang it! Your period ticket for the local Bus-Go-On-system (BGO) has expired. At first you wanted to buy a new period already today, but you suddenly realize that your next ticket would then expire a few days before your vacation starts, leaving a few trips you need to pay for individually anyways. Perhaps it is cheaper to pay for an individual trip now, allowing the next period ticket to cover more trips of the future?

Pixabay License, by TuendeBede via Pixabay

Input

The first line of input contains four positive integers s, p, m, and n. s ($1 \leq s \leq 10^9$) is the price of a single trip with the BGO, p ($1 \leq p \leq 10^9$) is the price of a period ticket, m ($1 \leq m \leq 10^9$) is the number of days the period ticket will cover, and n ($1 \leq p \leq 10^6$) is the number of trips you plan to make in the future (until you die, and hence won't need to pay for any further trips).

On the second line follows n non-negative integers in non-decreasing order, $t_1, t_2, \ldots, t_n$, where t_i ($0 \leq t_i \leq 10^9$) is the number of days until you will make your i-th trip with the BGO.

Output

The smallest possible cost of making the trips.

Sample Input 1	Sample Output 1
10 25 30 6 0 1 2 30 30 32	45

This page is intentionally left blank.

Problem C
Climbing Stairs
Problem ID: ../climbingstairs

CC-BY-SA 4.0, Inner staircase of the Baron Empain Palace tower in Cairo, Egypt, by Manadily via Wikimedia Commons

Your workplace recently announced a staircase cup with some fabulous prizes to win. The rules of participation are simple: Each day, it is possible to gain one point, and at the end of the month, whoever has the most points wins. In order to get a point for a specific day, you need to first walk n steps (either upwards or downwards) in the staircase — thereafter you must register your achievement in the registration book at the registration desk.

While you would like to participate in the cup (the prizes are quite nice) you also don't like walking on stairs that much, and would like to know the minimal number of steps you need to walk each day in order to get a point. You also plan to spend most of your day in your office, and you will enter and leave the building through the only entrance on the ground floor. There are no elevators and no underground floors in the building.

Input

One line with three numbers n ($1 \leq n \leq 1\,000$), r ($0 \leq r \leq 1\,000$) and k ($0 \leq k \leq 1\,000$), respectively indicating the number of steps required before you can register, the number of steps from the ground floor to the registration desk, and the number of steps from the ground floor to the floor of your office.

Output

The minimal number of staircase steps needed to participate in the staircase cup each day.

Sample Input 1	Sample Output 1
20 10 5	30

This page is intentionally left blank.

Problem D
Drive Safely
Problem ID: ../drivesafely

CC0 via Pxhere

A new stretch of road has just been built from Askøy to Bergen. Before the road can be opened for the public, however, a decision needs to be made as for which speed limits to impose. The minister of transportation wants the travelling time from Askøy to Bergen to be as small as possible, but he also imposes some constraints:

- For safety reasons, the speed limit through a turn of α degrees can be at most $|180 - \alpha|$ km/h. Even on a straight line, the speed limit can not exceed 180 km/h.
- To save money on sign costs, the minister of transportation will only allow k speed limit signs to be placed along the road.

The road is designed as a *polyline*, a sequence of n coordinates: the road starts in the first location, follows a straight line to the second location, from there follows a straight line to the third location, and so forth, until reaching the final location. Note that the road might cross itself with bridges or tunnels, but there will not be any intersections where a car can take a shortcut.

The speed limit is initially 180 km/h. Note that it is allowed to place two speed limit signs very close to each other, and the minister of transportation will allow speed limit signs to hold decimal values with infinite precision.

Input

The first line of input contains two positive integers n ($2 \leq n \leq 200$), and k ($1 \leq k \leq 50$). On each of the next n lines follows the locations of the polyline describing the road, starting with where the road starts at Askøy, ending with where the road ends in Bergen. The i^{th} location is described by two real numbers x_i and y_i ($-1\,000\,000 \leq x_i, y_i \leq 1\,000\,000$), denoting the coordinates in kilometers away from the arbitrarily chosen origin. All turns are at most 179 degrees either clockwise or counterclockwise, and coordinates are given with at most 6 digits after the decimal point.

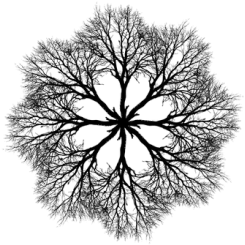
Output

A single real number, the shortest possible travelling time from Askøy to Bergen (in hours). Any answer within an absolute or relative error of 10^{-6} will be accepted.

Sample Input 1	Sample Output 1
6 2 0.0 0.0 90.0 0.0 90.0 30.0 120.0 30.0 120.0 0.0 210.0 0.0	2.0

Problem E
Equilibrium
Problem ID: ../equilibrium2

There are city elections in Treetopia! Treetopia, as you should know, is a quite unique country; there is *exactly* one way of travelling between *every* pair of cities! Two cities are said to be neighbouring if it is possible to travel from one to the other without visiting any other cities along the way, and the relationship between neighbouring cities is really something special.



Pixabay License, by Gerd Altmann, via Pixabay

The elections result are now being counted, and in just a short while the results will be announced on public broadcasting. This year, an election observer is standing ready in every of Treetopia's n cities to report on any problems they find. You know that every observer is very particular about the *order* that the results are announced. In particular, an observer in city i will count how many of the i^{th} city's neighbours are announced *before* city i (denoted b_i) as well as how many of its neighbouring cities are announced *after* city i (denoted a_i).

The observer will expect a_i to equal b_i. In fact, for every number the two numbers differ, $|a_i - b_i|$, the observer will send an angry letter of complaint. Desperate to avoid mountains of useless mail, you wonder which ordering you should choose to minimize the total number of complaints received.

Input

The first line of input contains a positive integer n ($1 \leq n \leq 100\,000$), the number of cities in Treetopia. Then follows $n - 1$ lines, the i^{th} of which contains two distinct integers u_i and v_i ($0 \leq u_i, v_i < n$), indicating that u_i and v_i are neighbouring cities.

Output

A single line with n space-separated integers, representing an order of the cities in Treetopia such that announcing the election results in this order minimize the number of received complaints. If there are several optimal orders, you can output any of them.

Sample Input 1	Sample Output 1
3 0 1 0 2	2 0 1

This page is intentionally left blank.

Problem F
Fence Bowling
Problem ID: ../fencebowling

Olav is spending an evening on his own, practicing in the bowling hall. Annoyingly, the side-rails in his lane are stuck in the active position, so if the ball goes out of bounds, then it simply bounces back in. This seems unfair to Olav, so he decide that any throw that does not bounce exactly k times in the fences before it hits the pins will be disqualified.

For this purpose, Olav has studied in detail how the ball bounces. His finding is that a ball hitting the fence with angle α relative to the normal, will bounce back with degree

$$\arctan(2\tan(\alpha))$$

relative to the normal. See the example in the picture to the right.

With which angle β relative to the normal against the rails should Olav throw the ball to hit a strike after first bouncing k times? Note that to hit a strike, Olav must hit *exactly* the middle of the bowling lane when the ball reaches the end of the lane. He always begins the throw from the middle of the lane as well.

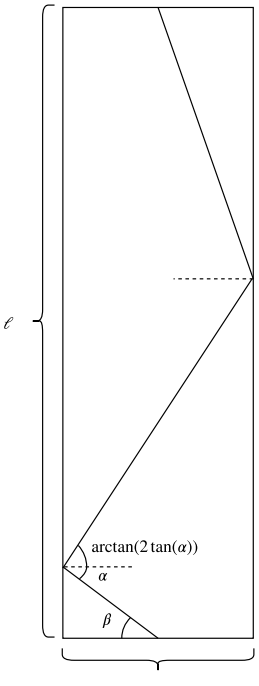


Illustration of the case $k = 2$

Input

The first and only line of input contains three positive integers k, w and ℓ. Here, k ($1 \leq k \leq 10$) is the number of bounces required, w ($1 \leq w \leq 100$) is the width of the bowling lane, and ℓ ($1 \leq \ell \leq 100$) is the length of the bowling lane.

Output

A single real number, the angle β in degrees. Any answer within an absolute or relative error of 10^{-6} will be accepted as a correct answer.

Sample Input 1	Sample Output 1
2 8 27	36.8698976

This page is intentionally left blank.

Problem G
Great GDP
Problem ID: ../greatgdp

Pixabay License, by Timofey Iasinskii/waiguobox

In your beloved homeland Treetopia, there is exactly one way of travelling between any pair of cites. In the loathful Cyclostan, on the other hand, there are exactly *two* ways of travelling between every pair of cities.

A delegation from Cyclostan is coming to visit Treetopia, and you realize that this is your big chance of persuading them to adopt the benefits of a tree-like society! Your friend Øyvind will decide on a travel plan for the Cyclostan delegation. In order to impress the delegation as much as possible, you persuade Øyvind to take them to parts of the country such that the GDP per capita is maximized across the visited cities. The trip can include visiting cities on several branches in the country, and it is not possible to travel through a city without visiting it.

The one and only airport in Treetopia is in the capital Treetopolis, and this is where the delegation from Cyclostan will arrive.

Input

The first line of input contains an integer n ($1 \leq n \leq 100\,000$), the number of cities in Treetopia. Then follows a line with n non-negative integers $c_1, c_2, \ldots c_n$ ($0 \leq c_i \leq 1\,000\,000$ for each $i \in \{1, 2, \ldots, n\}$), the GDP of each city. Then follows a line with n positive integers $k_1, k_2, \ldots, k_n$ ($1 \leq k_i \leq 1\,000\,000$ for each $i \in \{1, 2, \ldots, n\}$), the population of each city in Treetopia. Then follows $n - 1$ lines, the j^{th} of which with two distinct integers u_j and v_j ($1 \leq u_j, v_j \leq n$), indicating that there is a road between cities u_j and v_j. Treetopolis is city number 1.

Output

The highest possible GDP per capita of a connected region of Treetopia that contains Treetopolis. Any answer within an absolute or relative error of 10^6 will be accepted as a correct answer.

Sample Input 1	Sample Output 1
3 3 10 40 1 2 10 1 2 1 3	4.33333333333

This page is intentionally left blank.

Problem H
Howl
Problem ID: ../howl

Pixabay License, Alexas Fotos

Returning to the Beautiful Gloomy Outback (BGO) after a strenuous trip to the city, you hear a faint howl in the distance. Instantly you realize it is your friend Fenrir inviting you to a howling contest.

In order to make an impressive howl that wins the contest, you know that your howl must fulfill the following criteria in order to be valid:

- It must consist of a combination of the letters `A`, `H`, `O` and `W`. Each letter must occur at least once.
- The howl can not contain two consecutive `W`'s, or two consecutive `H`'s.
- The howl can not contain an `H` followed immediately by a `W` or an `A`.
- There can never be an `A` after the first occurrence of an `O`.

Can you produce a longer howl than Fenrir and win the contest?

Input

The first and only line of input contains a single word, the howl of Fenrir. Since Fenrir is a proper wolf, his howls are always valid. Writing down Fenrir's howl will require at most 1 MB of computer memory.

Output

A valid howl which will win the howling contest.

Sample Input 1	Sample Output 1
AAHOOW	AWAWHOO

This page is intentionally left blank.

Problem I
Ice Cream
Problem ID: ../icecream2

Public Domain, U.S. Air Force photo by Machiko Arita

The Better Get Obese (BGO) ice cream factory is gearing up for the holiday season, and after several years with mediocre sales, they have decided to focus only on their most popular product, ice cream with flavour of chocolate and vanilla.

In order to make a perfect such ice cream, it is important that the mixing machine receives equal amounts of vanilla and chocolate ice cream.

There are two separate creaming machines in the factory which produce respectively chocolate and vanilla ice cream, and the resulting creamy goodness is stored in two separate tanks. From there, it can be transported to the mixing machine through pipes, however the dimension of a pipe gives an upper bound for how much ice cream can pass through it each minute. These pipes meet in welding points, where streams of ice cream go from one pipe to another. Streams can also merge or split into other streams at such points, if more than two pipes meet here. It is not important to keep the flavors separate during the transportation, since they will eventually be mixed anyways.

Given a map of the pipe system, can you decide how many liters of ice cream the factory can produce each minute?

Input

The first line of input contains two integers n ($3 \leq n \leq 200$) and m ($2 \leq m \leq 1\,000$), respectively the number of welding points, and the number of pipes. The second line contains three distinct integers f, c and v ($1 \leq f, c, v \leq n$), the welding points where respectively the mixing machine, the chocolate tank and the vanilla tank is connected to the pipe system.

Finally follows m lines, the i^{th} of which contains three non-negative integers u_i, v_i and x_i ($1 \leq u_i, v_i \leq n$, $1 \leq x_i \leq 1\,000$). These indicate that there is a pipe between welding points u_i and v_i with a capacity for transporting x_i liters of ice cream per minute. Note that several pipes can go in parallel between the same welding points.

Output

A single integer, the maximum amount of ice cream the BGO factory can produce each minute (in liters).

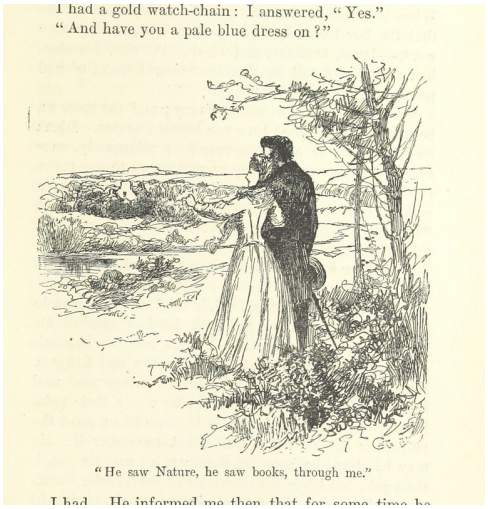
Sample Input 1	Sample Output 1
3 2 2 1 3 1 2 2 3 2 3	4

This page is intentionally left blank.

Problem J
Jane Eyre
Problem ID: ../janeeyre

Anna wants to read the famous book *Jane Eyre*, but annoyingly its title is somewhat late in the alphabet. This is a problem, since Anna always reads books in alphabetical order; as soon as she finish reading a book, she immediately begins reading the next book in her possession which comes first according to ASCII-order.

To make matters even worse, Anna often receives new books as presents. Such books go into the pile of Anna's unread books (she will finish the book she is currently reading even if the received book is earlier in the alphabet). If she receives one or more books at the exact same moment as she finishes another book, though, then she will pick her next book among both the books in her existing pile and the newly received books.



Given Anna's pile of unread books and a schedule for at which points in time Anna's friends will give her new books, can you figure out when she will finish reading *Jane Eyre*? Anna reads at a speed of one page per minute.

Input

On the first line are three non-negative integers n, m, and k; here n ($0 \leq n < 100\,000$) indicates the number of unread books in Anna's pile (in addition to *Jane Eyre*), m ($0 \leq m < 100\,000$) indicates the number of books her friends will give her, and k ($1 \leq k < 100\,000$) indicates the number of pages in *Jane Eyre*.

The next n lines describe the other unread books in Anna's pile; the i^{th} such line contains a string s_i ($1 \leq |s_i| \leq 20$) and a positive integer k_i ($1 \leq k_i < 100\,000$) indicating respectively the title of the book and how many pages it contains. The string s_i will be enclosed in double quotes (`"`), and consists of a mixture of spaces and alphanumeric ASCII characters.

Finally follows m lines describing the books Anna's friends will give her; the j^{th} such line contains a non-negative integer t_j ($0 \leq t_j \leq 1\,000\,000\,000$), a string s_j ($1 \leq |s_j| \leq 20$) and a positive integer k_j ($1 \leq k_j < 100\,000$) indicating respectively the time (in minutes from now) Anna will receive the book, the title of the book and how many pages it contains. The string s_j will be enclosed in double quotes (`"`), and consists of a mixture of spaces and alphanumeric ASCII characters.

Output

A single integer, the minute at which Anna finish reading *Jane Eyre*.

Sample Input 1	Sample Output 1
2 2 592 "Pride and Predjudice" 432 "Don Quixote" 863 863 "Great Gatsby" 218 1082 "Crime and Punishment" 545	1673

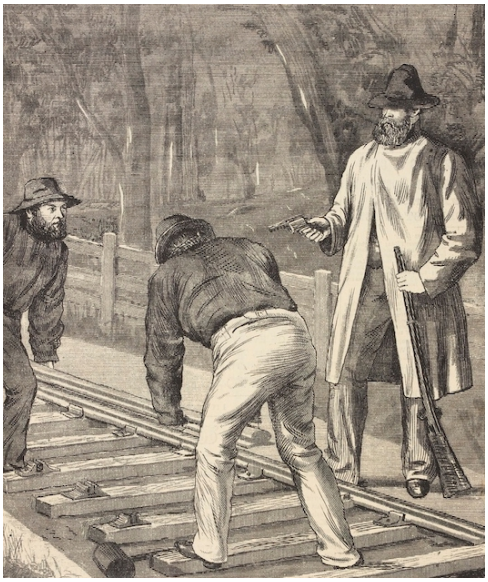
Problem K
Killing Chaos
Problem ID: ../killingchaos

In the dangerous wild west, robbers are attacking a long train with many coaches. Chaos erupts, and the robbers realize that the amount of chaos equals the number of passengers in the train, rounded up to the nearest multiple of 10. In order to quench the chaos, they therefore decide to kill some passengers by blowing up one of the coaches.

What the robbers failed to realize, however, is that when there are multiple disjoint train segments, then the total amount of chaos is equal to the sum of the chaos of each train segment *times* the number of train segments!

Frantic to quench the now even worse chaos, the robbers continue blowing up every coach until all passengers are dead. Phew!

The chaos in a train segment is equal to the number passengers in that train segment rounded up to the nearest multiple of 10. What was the maximum amount of chaos during the robbery?



Public Domain, via Wikimedia Commons

Input

On the first line is a single integer n, ($3 \leq n \leq 100\,000$), the number of coaches in the train. On the second line follows n integers $p_1, p_2, \ldots p_n$, ($0 \leq p_i \leq 100$ for each $i \in \{1, 2, \ldots, n\}$) the number of passengers in each coach. On the third and final line follows a permutation of the numbers from 1 to n indicating the order in which the robbers blew up the coaches.

Output

A single integer, the maximum chaos that occurred during the robbery.

Sample Input 1	Sample Output 1
5 3 5 10 2 5 2 4 5 1 3	90

Sample Input 2	Sample Output 2
4 32 3 3 3 1 3 2 4	50

This page is intentionally left blank.